THIS PAGE BLANK (USPTO)

(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 0 957 447 A2

(12) EUROPEAN PATENT APPLICATION

(43) Date of publication:
17.11.1999 Bulletin 1999/46

(51) Int. Cl.^{6}: G06K 15/00

(21) Application number: 99109615.7

(22) Date of filing: 14.05.1999

(84) Designated Contracting States:
AT BE CH CY DE DK ES FI FR GB GR IE IT LI LU MC NL PT SE
Designated Extension States:
AL LT LV MK RO SI

(30) Priority: 14.05.1998 JP 13207298

(71) Applicant:
SEIKO EPSON CORPORATION
Shinjuku-ku, Tokyo 163-0811 (JP)

(72) Inventor: Teradaira, Mitsuaki
Suwa-shi, Nagano-ken 392-8502 (JP)

(74) Representative:
Hoffmann, Eckart, Dipl.-Ing.
Patentanwalt,
Bahnhofstrasse 103
82166 Gräfelfing (DE)

(54) **Printer and control method of saving data from volatile to nonvolatile memory in the printer**

(57) A printer comprises nonvolatile memory means (5) having a plurality of memory areas (A1 -A6); volatile memory means (3) having a plurality of memory areas (A1-A6), each of said plurality of memory areas of said volatile memory corresponding to a respective one of said plurality of memory areas of said nonvolatile memory; control means (56) for detecting each occurrence of each of a plurality of predetermined trigger events, for generating a save command in response to each detected trigger event, and for designating one or more of said memory areas (A1-A6) of said volatile memory (3) in accordance with the particular trigger event detected; and saving means (51, 52, 54) responsive to said save command for saving data stored in the designated one or more memory areas of said volatile memory (3) to the corresponding one or more memory areas (A1-A6) of said nonvolatile memory (5).

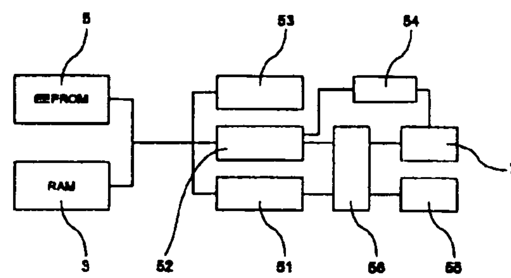

FIG.1

EP 0 957 447 A2

Description

[0001] Th present invention relates to a printer having a nonvolatile memory and a volatile memory, both for storing information indicative of the operating condition of the printer. During printer operation, the information is initially copied from the nonvolatile memory to the volatile memory. The information in the volatile memory is then used for printer control and is updated whenever required. The updated information is saved to the nonvolatile memory. The invention also relates to a control method of writing of the updated information back into the nonvolatile memory.

[0002] Correct printer operation requires certain information about the operating conditions of the printer, i.e., status information, such as the remaining ink quantity and the last head cleaning time (if it is an ink jet printer), the print head location, and how much roll paper remains (if the printer is one that prints on roll paper). To explain "head cleaning time", it is well known that ink jet printers, which print by ejecting ink from nozzles of an ink jet head onto a printing medium, require regular maintenance, that is, cleaning of the nozzles, in order to sustain reliable operation. Such regular cleaning is needed to prevent problems such as clogged nozzles resulting from ink drying and becoming viscous inside the nozzles. This cleaning process is generally managed based on a timer and is implemented with different levels depending on how much time has passed since the last cleaning operation. Storing, in case of an ink jet printer, the last head cleaning time and level as pieces of the status information mentioned above enables the next cleaning operation to be more appropriately performed.

[0003] The status information further includes count values such as the printed character count, the accumulated paper feed amount, and the total operating time.

[0004] When the printer stops in the middle of a printing job due to some error or other event like turning off of the power switch, a power failure or a printer reset, the knowledge of such status information becomes particularly desirable so that the operating conditions of the printer at the time when the interruption occurred can be determined and an appropriate process for recovering from the interruption executed. In order to keep such status information even under no-power conditions, printers have commonly comprised an EEPROM (Electrically Erasable Programmable ROM), a flash memory or other type of nonvolatile memory. Printer operating conditions such as those described above (also referred to as "status data" below) are maintained and updated in a volatile memory such as a DRAM, and saved to the nonvolatile memory at a regular saving period. Each time the printer resumes operation, the status data is copied from the nonvolatile memory to the volatile memory.

[0005] As noted above, status data is written into the nonvolatile memory at periodic time intervals during the operation of the above-described conventional printer. More specifically, every time a predetermined interval (the saving period), that is measured by an internal timer elapses, any current printer process is interrupted, and the printer CPU is diverted to the process for writing (saving) the data into the nonvolatile memory. The ability of the printer to recover from printing interruptions and restore the operating status to that before the interruption is thus improved by regularly saving the status data to the nonvolatile memory.

[0006] It is obviously not possible to predict when an error or other interruption is going to occur. This makes it desirable for the saving period to be as short as possible in order to assure that the most recent condition of the printer before an interruption occurs is saved with the greatest possible accuracy.

[0007] When the CPU is executing a printing operation or otherwise processing data and is interrupted by the saving process for writing into the nonvolatile memory, the interrupted CPU task (printing or data processing in this example) stops until the writing into the nonvolatile memory is completed. Printer throughput therefore drops. As the amount of data being tracked and saved to the nonvolatile memory increases, the saving time obviously also increases, and can increase to a level at which the drop in printer throughput can no longer be ignored. It is therefore desirable that the saving time, i.e., the time required for copying data from the volatile memory to the nonvolatile memory, at each saving instance be as short as possible irrespective of the amount of status data to be saved.

[0008] Conventional saving methods copy the entire content of the volatile memory into the nonvolatile memory whenever data saving is initiated, regardless of whether all data in the volatile memory have indeed changed since they were last saved. This means that time may be spent for saving data that have not changed and, thus, need not be saved. In addition, nonvolatile memories allow a limited number of write cycles to a particular memory location. Thus, the service life of nonvolatile memories depends upon the number of write cycles they undergo per unit time. This means that when data that has not changed is written to the nonvolatile memory, the limited number of available write cycles is needlessly reduced, and memory service life is potentially shortened.

[0009] An object of the present invention is therefore to shorten the saving time and thereby reduce the effect the saving process has on the normal printer operation. A further object of the present invention is to prevent data from being unnecessarily saved to the nonvolatile memory, and thereby improve the service life of the nonvolatile memory.

[0010] These objects are achieved with a printer as claimed in claim 1 and a control method as claimed in claim 6. Preferred embodiments of the invention are subject-matter of the dependent claims.

[0011] By logically dividing both the volatile memory and the nonvolatile memory into a number of memory areas and saving the corresponding data groups of different ones of these memory areas in response to different ones of prede-

termined trigger events, the saving time is shortened and the write frequency for the individual memory areas is reduced so that life time of the nonvolatile memory is improved.

[0012] Trigger events are events in response to which data is saved from the volatile memory to the n nvolatile memory. These trigger events preferably include detection of a reset signal from a host computer. In this case, the printer is reset based on this reset signal after data have been saved to the nonvolatile memory. Other trigger events may include events such as: the periodic lapse of a specific time interval (the saving period), turning printer power on and off, and specific control events in printer operation. These specific control events preferably include print head cleaning, ink cartridge replacement, roll paper replacement, and such operating errors as an increase in the print head temperature beyond a threshold value or a disconnection or rupture of the carriage transportation belt. It is to be noted that these specific control events depend on the type of printer to which the invention is applied. Thus, control events like print head cleaning and ink cartridge replacement apply to ink jet printers only.

[0013] According to a preferred embodiment of the invention, some or all of data groups corresponding to the memory areas of the volatile memory and the nonvolatile memory are classified into at least a first and a second set of data groups each set including one or more of the data groups. The data group or groups belonging to the first set is or are saved periodically with the saving period. The data group or groups belonging to the second set is or are saved in response to a predetermined one of the control events.

[0014] Much of the data to be saved to the nonvolatile memory is updated in the volatile memory when the printer performs a particular control operation when particular control events such as those described above occur. It is therefore possible to minimize damage from data loss by saving those data to the nonvolatile memory after the control event has been completed and the data in the volatile memory updated. By also saving data that is otherwise periodically saved and also restarting the time counter measuring the saving period, an increase in the frequency of data write cycles to the nonvolatile memory can be prevented.

[0015] Other objects and attainments together with a fuller understanding of the invention will become apparent and appreciated by referring to the following description of preferred embodiments taken in conjunction with the accompanying drawings, in which:

Fig. 1 is a typical functional block diagram of a printer according to a preferred embodiment of the present invention;

Fig. 2 is an example of circuit block diagram suitable for implementing the functions shown in Fig. 1;

Fig. 3 shows an example of the logical structure of memory areas in the nonvolatile memory and the volatile memory;

Fig. 4 is a table showing an example of the relationship between data groups and trigger events that trigger saving of the data groups from the volatile memory to the nonvolatile memory;

Fig. 5 is a flow chart illustrating a general saving process according to the present invention;

Fig. 6 is a flow chart of a first saving process according to a preferred embodiment of the present invention;

Fig. 7 is a flow chart of an second saving process according to a preferred embodiment of the present invention; and

Fig. 8 shows an example of a preferred logical structure of memory areas A5 both in the nonvolatile memory and the volatile memory.

[0016] The printer 1 shown in Fig. 1 is - by way of example only - a serial ink jet printer capable of printing to both cutsheet forms and roll paper. This printer 1 comprises an EEPROM 5 for storing status data, a RAM 3 for temporarily storing status data as it is updated during printer operation, various control means 51 to 56 for controlling the copying (saving) of the status data from the RAM 3 to the EEPROM 5 and from the EEPROM 5 to the RAM 3, and a counter 7.

[0017] Both memories, the EEPROM 5 and the RAM 3, (or a respective part of each) are logically divided into a plurality of memory areas for storing status data such those explained above. The status data is in turn divided into a corresponding plurality of data groups and each data group is assigned to a respective one the of memory areas. As will be described in further detail below, the individual data groups are saved at different times.

[0018] As shown in Fig. 3 and Fig. 4, in this embodiment the memories (the memory parts used for storing status data) ar l gically divided into six memory areas A1 to A6 each, and the status data are correspondingly grouped into six data groups. These six data groups are preferably classified into a number of sets. Among these are at least a first

set of one or more data groups that is or are periodically saved to the nonvolatile memory at a predetermined saving period as counted by counter 7, and a second set of one or more data groups that is or are saved to the nonvolatile memory every time a predetermined control event (or one of multiple specific control events) occurs. To distinguish from the periodic saving of the first set, saving of the second set is also referred to as "event-driven" in this text. Despite this distinction, the term "trigger event" as used in this text includes both the occurrence of a signal from counter 7 indicating lapse of the saving period and triggering the periodic saving, as well as the occurrence of a control event. As will be appreciated from the following discussion, while the first and second sets are different, they may overlap in the sense that one or more data groups may belong to both sets. As will also be appreciated from the following discussion, in the described embodiment there are indeed five different sets of data groups. Since for an understanding of the principle of defining and using different sets of data it should be sufficient to refer to the first and second sets, the remaining sets will not be discussed in further detail below.

[0019] When the saving period counted by counter 7 elapses, a controller 56 issues to a first read/write means 51 a command for copying the first set of data groups from RAM 3 to EEPROM 5.

[0020] The control events that trigger saving of the second set of data groups include, for example, the print head cleaning. When a print head cleaning event occurs, for example, a monitoring means 55 refers to a timer to determine whether the time that has elapsed since the last time the print head was cleaned exceeds a predetermined length of time. If it does, the controller 56 issues to a second read/write means 52 a command for copying the second set of data groups from RAM 3 to EEPROM 5. A command for copying the first set to EEPROM 5 is also issued at this time so that both the first and second sets of data groups are saved to EEPROM 5.

[0021] An initialization means 54 re-initializes counter 7 whenever EEPROM 5 is written by second read/write means 52. Note that when the printer power is turned on, third read/write means 53 loads the data from EEPROM 5 into RAM 3.

[0022] It should be further noted that, in the present embodiment, the above-noted first read/write means 51, second read/write means 52, third read/write means 53, initialization means 54, and controller 56 are implemented by means of a program-controlled microprocessor and storage means storing the program code for the respective operations. This storage means could be, for example, a ROM or a temporary storage such as a RAM (including part of RAM 3). The monitoring means 55 is also preferably implemented by a program-controlled microprocessor, a ROM or other storage for storing the program code for the respective operations, a timer, and a sensor (not shown in the figures) for detecting a change in printer operation. Preferably, the microprocessor and ROM or other storage used for implementing first read/write means 51 and others are the same as those used for implementing the monitoring means 55.

[0023] Fig. 2 is a circuit block diagram of a printer according to this first embodiment of the present invention. As shown in Fig. 2, this printer 1 comprises a CPU 2 (representing the microprocessor mentioned above) for controlling the overall printer operation, the RAM 3 as main work memory, a ROM 4 for storing control data and a control program, the EEPROM 5 for storing status data as described above, a print mechanism 6 for printing on a printing medium by means of a print head, the counter 7 for counting the passage of time and an interface 8 for connection to a host computer.

[0024] The CPU 2 loads program and control data from ROM 4 and status data from EEPROM 5 into RAM 3 and controls printer operation based on the loaded program and control data in RAM 3. When power to printer 1 is turned on, the printer is initialized based on the loaded data. The printer (CPU 2) then waits for a command from the host.

[0025] The printer 1 has various sensors, not shown in the figures, for monitoring the operating conditions of the printer. The status data in RAM 3 which is generated by and in response to these sensors, respectively, is sequentially updated, and in response to specific trigger events described in detail below it is copied and saved to specific memory areas in EEPROM 5. The status data saved to EEPROM 5 is loaded from EEPROM 5 into RAM 3 when the printer 1 is initialized or reset, and is thus used as the initial data for the printer control.

[0026] For example, the head cleaning time described above represents the time of the last head cleaning event. The CPU 2 uses this head cleaning time to determine the timing of the start of the next head cleaning event, and initiates head cleaning at that time.

[0027] In addition, when a printer error occurs, causing the printer to stop operating in the middle of a print job, for example, the printer 1 is restored to the previous normal operating condition based on the location of the print head read from EEPROM 5 after the printer 1 is reset. The timing of saving data to the EEPROM 5 and the saving procedure itself as well as the copying from the EEPROM 5 to the RAM 3, are controlled by the CPU 2.

[0028] The counter 7 measures the elapse of a predetermined saving period based on an internal clock circuit (not shown in the figures). Whenever data is saved to EEPROM 6, the CPU 2 re-starts counter 7 to count said saving period. When this predetermined period elapses (1 hour, for example), the counter 7 outputs a signal notifying the CPU 2 that the counted period has elapsed. In response to this signal the CPU 2 acts to save the above mentioned first set of data groups to EEPROM 5. It goes without saying that the counter 7 may also be implemented by the CPU 2.

[0029] The CPU 2 also has a reset signal detection function 9 and a reset function 10. This reset signal detection function 9 detects when a reset signal has been received from the host computer. When a reset signal is detected, the CPU 2 issues a write command for saving at least part of the data in RAM 3 t EEPROM 5. The reset function 10 is a

function for resetting the printer, an operation that includes copying all of the data in EEPROM 5 to RAM 3, returning various printer components (such as the print head) to respective initialization positions, and entering a standby state waiting for data from the host. When data saving to EEPROM 5 initiated by a reset signal is completed, the CPU 2 performs the reset function 10. This sequence ensures that data relating to the printer status before the reset are saved to EEPROM 5.

[0030] Referring to Figs. 3 and 4, the memory areas of the EEPROM 5 and the RAM 3 in the preferred embodiment will be explained in more detail. Fig. 3 is an illustration of an example of the logical structure of the memory areas in EEPROM 5 and RAM 3 used for storing status data. As mentioned before and shown in the figure, the EEPROM 5 and RAM 3 (or part of them) are logically segmented into six memory areas A1 to A6. Each of the areas in EEPROM 5 corresponds to an area in RAM 3, enabling data to be copied in units of a memory area from RAM 3 to EEPROM 5 and vice versa. As noted above, the status data are also divided into six data groups. These six data groups are each assigned to a respective one of the six memory areas. As a result, data storage is accomplished in units of data group, copying one or a plurality of data groups at a time.

[0031] Each of memory areas A1 to A6 includes data area and a checksum area. When data is copied between EEPROM 5 and RAM 3, the checksum of each copied data group is used for error detection. Each time a data group is copied from a source memory area (in EEPROM 5 or RAM 3) to a destination memory area (in RAM 3 or ESPROM 5) the checksum is calculated based on the copied data group in the destination memory area and compared with the checksum stored for that data group in the source or destination memory area.

[0032] Default (initialization) values for the status data are stored in ROM 4 or in another memory area (other than memory areas A1 to A6) of EEPROM 5 in Fig. 2. These default values are values suitable to assure normal printer operation. When the checksum comparison explained above reveals a data error for a data group, the status data stored in the related pair of memory areas of RAM 3 and EEPROM 5 are replaced with the default values. This makes it possible to minimize printer operating problems when an error occurs.

[0033] Fig. 4 is a table showing the relationship between data groups A1 to A6 and the trigger events upon which the respective data groups are saved to EEPROM 5. One of the first steps in memory control according to the present invention is to select a plurality of "trigger events" at which certain data groups are saved to the respective memory areas. As shown in Fig. 4, these trigger events include in the present embodiment: turning on power to the printer, lapse of the saving period (such as 1 hour), specific printer events (such as print head cleaning, ink cartridge and roll paper replacement, occurrence of a printer operating error), the moment when a counter for tracking specific control information indicates a change of that control information by a predetermined amount (e.g., the count value of a counter for tracking the accumulated drive count of a roll paper cutter reaches a specific value), and detection of a reset signal (including reset signals from the host and a reset signal generated when an abnormal voltage is detected).

[0034] As described above, in the present exemplary embodiment of the invention the status data are divided into six data groups according to the trigger events described above, and assigned to the six memory areas shown in Fig. 3. In this preferred embodiment, the following status data are included in the respective data groups (note that the data groups are given the same names A1 to A6 as the corresponding memory areas).

Data Group/Memory Area	Status Data
A1	time at which power to the printer is turned on
A2	printed line count since the last print head cleaning event, and total printed line count since printer manufacture
A3	time of the last print head cleaning event, and the cleaning level
A4	cumulative drive count of the roll paper cutter
A5	reset flag, time of the most recent trigger event, ink consumption counter
A6	various complementary data (such as carriage travel distance)

[0035] According to the example of the relation between trigger events and data groups shown in Fig. 4, when the power to printer 1 is turned on, data group A1, i.e. the current time is saved to EEPROM 5, and data groups A2 and A5 are saved periodically with the saving period. When any of the specific control events occurs, data groups A2 t A5 are saved; and when a reset occurs, only data group A5 is saved. Each time a counter for counting particular contr l information has changed by a predetermined amount or reaches a predetermined value, data groups A4 and A5 are saved.

Data group A6 stored in memory area A6 is used, for example, to correct the printing position as a result of variations between specific mechanical components of the printer, and is written prior to shipping from the factory and when the printer is repaired. It will thus be obvious that not all data groups need to b saved at every trigger event shown in Fig. 4.

[0036] While the first and the second set of data groups have been referred to above, it is to be noted that, as shown in Fig. 4, in the present embodiment there are three other sets, namely a third set including only data group A1, a fourth set of data groups A2, A4 and A5 and a fifth set of only data group A5.

[0037] It should be noted that the data group A5 includes information that is, relative to information of other data groups, important for the printer control, and is therefore more frequently saved. Furthermore, a reset signal is also generated when a power failure or a voltage drop is detected, for example. The data group A5 is therefore saved in response to a rest signal while the other data groups are not saved.

[0038] A table 40 (see Fig. 5) corresponding to that shown in Fig. 4 is stored in a memory area of ROM 4 or another memory area of EEPROM 5 and is read when any of the trigger events stored in the table occurs.

[0039] Fig. 5 is a flow chart of the saving process by which data is saved to EEPROM 5. This flow chart shows the procedure from when one of the trigger events shown in the table occurs until the data group(s) concerned is (are) copied from RAM 3 to EEPROM 5. When the CPU 2 detects any of the trigger events (step 401), the table 40 is read and used to determine the data group(s) to be saved to EEPROM 5 (step 402). The thus determined data group(s) is (are) then read out from RAM 3 and written to the corresponding area(s) in EEPROM 5 (step 403). As a result, only that data group(s) defined for being saved at a respective trigger event is indeed saved to EEPROM 5 when that trigger event occurs.

[0040] Fig. 6 and Fig. 7 are flow charts illustrating saving processes in a printer 1 according to a preferred embodiment of the present invention. Fig. 6 shows the periodic saving process (first saving process), and Fig. 7 shows the event-driven saving process (second saving process).

[0041] Referring to Fig. 6, data stored in EEPROM 5 is loaded into RAM 3 when the printer 1 is activated, and the counter 7 starts counting the saving period (step 201). When this saving period elapses after the counter 7 started counting (step 202), data groups A2 and A5 (forming the first set of data groups) are saved to EEPROM 5 (step 203) as mentioned above. If this interval is 1 hour, for example, these data groups are saved at least once every hour. When the saving of the data to EEPROM 5 is completed, the counter 7 is reset and begins counting the next saving period (the first saving process enters a standby mode in step 204). As a result, the data of data groups A2 and A5 is written to EEPROM 5 every time the predetermined saving period elapses.

[0042] As described above, data is also saved to EEPROM 5 at specific selected control events, that is, at trigger events other than that represented by the lapse of the saving period described above. As also noted above, exemplary control events include: print head cleaning, replacement of the ink cartridge or roll paper, and operating errors such as a rise in the print head temperature or disconnection of the carriage transportation belt. In response to such trigger event CPU 2 passes control to the second saving process shown in Fig. 7. More specifically, it is first detected when the control process, if any, related to the control event is completed (step 301), the control information (e.g., count values) that has been updated in RAM 3 by that control process is then saved to EEPROM 5 (step 302). As shown in Fig. 4, data groups A2 to A5 (forming the second set of data groups) are saved at this time. When the saving of data to EEPROM 5 is completed, counter 7 is reset (step 303). Resetting the counter 7 in this step causes the saving period counted in step 202 in Fig. 6 to restart at the completion of the process shown in Fig. 7.

[0043] The data group A3 is saved only in step 302 in Fig. 7. On the other hand, data groups A2 and A5 are saved to EEPROM 5 both periodically with the saving period (step 203 in Fig. 6), and together with data groups A3 and A4 when one of the specific control events occurs. Even though this may slightly increase the frequency of writing to the nonvolatile EEPROM, a drop in printer throughput is prevented because the counter 7 is reset in step 303.

[0044] Fig. 8 is an illustration of the logical structure of memory area A5 in EEPROM 5 and RAM 3 according to a preferred embodiment of the invention. As will be understood from the figure, memory area A5 in EEPROM 5 is further divided into a plurality of subareas A5A to A5N. Each of these subareas A5A to A5N includes, in addition to a status data area, a sequence area for storing a sequence value and a checksum area. When data group A5 is saved to EEPROM 5 it is written into a selected one of the subareas. The subareas are sequentially and cyclically selected. The sequence value is used to identify the subarea that holds the most recently saved data.

[0045] Referring to Fig. 5, let data group A5 be specified in step 402. In this case, when data group A5 is saved from RAM 3 to EEPROM 5 in step 403, it is written to the subarea next in sequence after the subarea last used. For example, if data group A5 was written to subarea A5A the last time step 403 was executed and data group A5 was specified, data group A5 is written to subarea A5B in the present step 403.

[0046] In this exemplary embodiment, a 2-byte area is allocated for the sequence value, enabling codes OO_H to FF_H to be stored, i.e., the sequence value can assume any value between 0 and 65536. Each time step 403 is executed, the sequence value is increased (since the sequence value changes cyclically it could be decreased instead) and the updated value is written into the sequence area of the respective subarea. For example, if data was written to subarea A5A the

[0047] last time step 403 was executed and a sequence value of 05 is stored in the sequence area of subarea A5A, a value of 06 is stored in the sequence area of subarea A5B in the present step 403. If the maximum sequence value (65536 in this embodiment) and the number of subareas are chosen to be different, th sequence value that will be stored in the sequence area of a particular subarea will always be different from that that was written into the same sequence area the previous time. Therefore, the sequence value stored in the subarea that holds the most recently saved data, and the sequence value stored in the subarea next in sequence to that subarea will be discontinuous. This discontinuity allows to identify among the subareas that one that holds the most recently saved data as well as those preceding.

[0048] When data is to be loaded from EEPROM 5 to memory area A5 of RAM 3 (which is not subdivided), the sequence values stored in subareas A5A to A5N are compared to find the above explained discontinuity, thereby to identify the subarea holding the most recent data and to load the most recent data into the memory area A5 of RAM 3. For example, if the values stored in the sequence areas of subareas A5A, A5B, and A5C are 05, 06, and FB, the status data stored in subarea A5B is the most recent one, and is therefore loaded into memory area A5 of RAM 3.

[0049] As shown in Fig. 4, the data group A5 is saved whenever a reset signal is detected, and is usually information that is important for the printer control. This information is also saved in response to other control events, so that the saving frequency for memory area A5 is greater than that of other memory areas. By providing a plurality of subareas in EEPROM 5 so that frequently updated data groups are written sequentially to different memory areas, it is possible to prevent a drop in life time of the EEPROM 5 resulting from a frequent use of a particular pan of that memory.

[0050] It will also be obvious that by providing a checksum area for storing a checksum as error detection code in each subarea, it is possible, when a data corruption is detected in the most recently saved status data, to scan and locate the second most recently saved status data and check for errors in this data. If there are no errors, the second most recently saved status data can then be loaded into RAM 3.

[0051] While the embodiment of the present invention has been described as using an EEPROM as the nonvolatile memory, a flash memory or other type of nonvolatile memory can be alternatively used. The data stored in the nonvolatile memory are of course not limited to that described above. More specifically, data relating to any printer condition can be stored, or only a subset of the above-described status data can be stored. In addition to storing status data, the nonvolatile memory can be used to store other data, including font data and program data.

[0052] The invention is also not limited to the trigger events described above. Furthermore, a checksum has been described for error detection, but it will be obvious that other error detection codes, including a parity check code, or even error detection and correction codes can be alternatively used.

Claims

1. A printer comprising:

   nonvolatile memory means (5) having a plurality of memory areas (A1-A6);
   volatile memory means (3) having a plurality of memory areas (A1 -A6), each of said plurality of memory areas of said volatile memory corresponding to a respective one of said plurality of memory areas of said nonvolatile memory;
   control means (56) for detecting each occurrence of each of a plurality of predetermined trigger events, for generating a save command in response to each detected trigger event, and for designating one or more of said memory areas (A1-A6) of said volatile memory (3) in accordance with the particular trigger event detected; and
   saving means (51, 52, 54) responsive to said save command for saving data stored in the designated one or more memory areas of said volatile memory (3) to the corresponding one or more memory areas (A1-A6) of said nonvolatile memory (5).

2. The printer as set forth in claim 1, wherein

   said control means comprises a counter (7) for counting a predetermined saving period, monitoring means for detecting the occurrence of a predetermined event relating to printer operation, and means for generating a first save command in response to said counter having counted said saving period and a second save command in response to said monitoring means having detected said event, and
   said saving means comprises first read/write means (51) for saving a first set of data comprising the data of one or more of said memory areas (A1-A6) in response to said first save command, second read/write means (52) for saving a second set of data comprising the data of one or more of said memory areas (A1-A6) in response to said second save command, wherein said second set of data includes the data f ne r more memory areas not included in said first set, and an initialization means for initializing said counter (7) in response to said saving by said second read/write means (52).

3. The printer as set forth in claim 1 or 2, wherein each memory area (A1-A6) comprises a data area for storing data representing printer operating conditions, and a checksum area for storing an error detection code.

4. The printer as set forth in any one of claims 1 to 3, wherein at least one (A5) of said plurality of memory areas (A1-A6) in said nonvolatile memory (5) is divided into a plurality of subareas (A5A - A5N) each subarea comprising a data area for storing data representing printer operating conditions and a sequence area for storing sequence data representing the sequence of use of the subareas.

5. The printer as set forth in claim 4, wherein each subarea (A5A - A5N) further comprises a checksum area for storing an error detection code.

6. A control method for saving data from the volatile memory (3) to the nonvolatile memory (5) in a printer (1) as defined in claim 1, said control method comprising steps of:

(a) grouping all data that require saving to the nonvolatile memory into a number of data groups,
(b) assigning each data group to a respective pair of corresponding memory areas of said volatile and said nonvolatile memories,
(c) defining specific different trigger events and allocating each of said data groups resulting from step (a) to one or more of said trigger events;
(d) detecting each occurrence of each of said trigger events and generating a save command in response to each detected trigger event,
(e) selecting among the data groups resulting from step (a) that or those allocated to the trigger event detected in step (d); and
(f) saving the data of the one or more data groups selected in step (e) to the corresponding one or more memory areas (A1-A6) of said nonvolatile memory (5).

7. The control method as set forth in claim 6, wherein said specific trigger events include the detection of a reset signal from a host computer to which the printer (1) is connected, and the printer is reset in response to this reset signal after step (f).

8. The control method as set forth in claim 6 or 7, wherein the specific trigger events include:

lapse of a predetermined saving period as a first trigger event, said saving period being periodically measured,
turning the power to the printer on or off as a second trigger event, and
specific control events occurring during printer operation as third trigger events.

9. The control method as set forth in claim 8, wherein said specific control events include print head cleaning.

10. The control method as set forth in claim 8 or 9, wherein

step (c) comprises allocating a first set of one or more data groups to said first trigger event and allocating a second set of one or more data groups to another trigger event, said second set including one or more data groups not included in said first set, and
step (f) comprises restarting measurement of said saving period each time the data of said second set have been saved in response to said other trigger event.

11. The control method as set forth in any one of claims 6 to 10 for saving data in a printer as defined in claim 3, wherein the control method further comprises the step of:

(g) computing an error detection code based on the data of a data group, comparing the computed error detection code with that one stored in the associated checksum area and indicating a data error if the compared error detection codes are different.

12. The control method as set forth in claim 11, further comprising a step of

(h) copying the data stored in the nonvolatile memory to the volatile memory under a predefined condition, wherein step (g) detects an error in the data of a memory area (A1-A6) read fr m the nonvolatile memory (5) during execution of step (h).

13. The control method as set forth in claim 6, wherein step (g) detects an error of a data area group after the data group is saved to the nonvolatile memory (5) in step (f).

14. The control method as set forth in any one of claims 11 to 13, further comprising a step of:

    (i) writing specific initialization data into the concerned pair of corresponding memory areas (A1-A6) in the volatile and the nonvolatile memories (3, 5) when a data error is indicated in step (g).

15. The control method as set forth in any one of claims 6 to 14 for saving data in a printer as defined in claim 4, wherein step (f) comprises:

    (f1) scanning the subareas (A5A - A5N) of the nonvolatile memory (5) and finding a pair a first one and a second one of two successive subareas for which the absolute difference between the values represented by the respective sequence data is higher than that of a predetermined positive or negative incremental value, thereby to identify said first subarea as the one among the subareas of said at least one memory area (A5) that holds the most recent data,
    (f2) reading the sequence data stored in the sequence area in said first subarea and adding said incremental value to the read sequence data so as to obtain updated sequence data, and
    (f3) writing the data group (A5) corresponding to said at least one memory area (A5) into the data area and said updated sequence data into the sequence area of said second subarea.

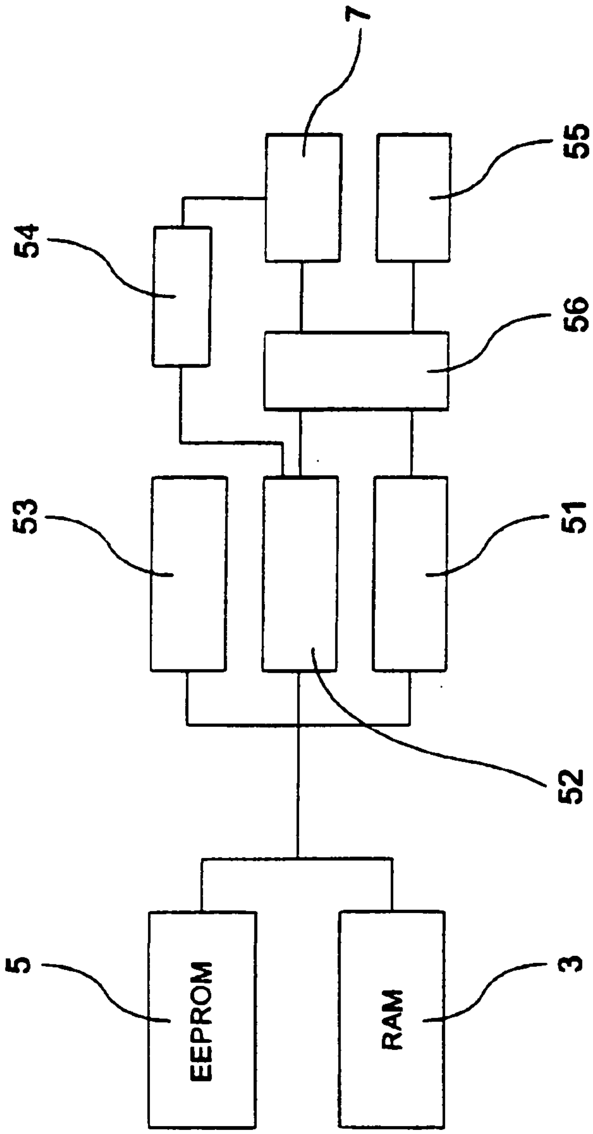

FIG.1

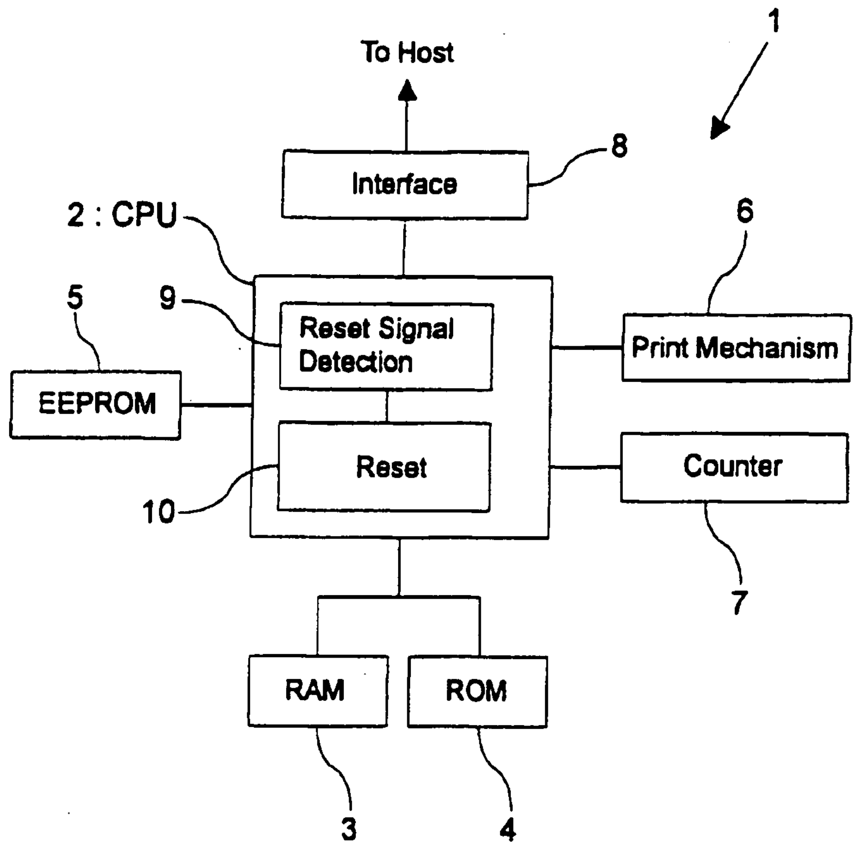
To Host
1
8
Interface
2 : CPU
6
5
9
Reset Signal Detection
Print Mechanism
EEPROM
Reset
Counter
10
7
RAM
ROM
3
4

FIG.2

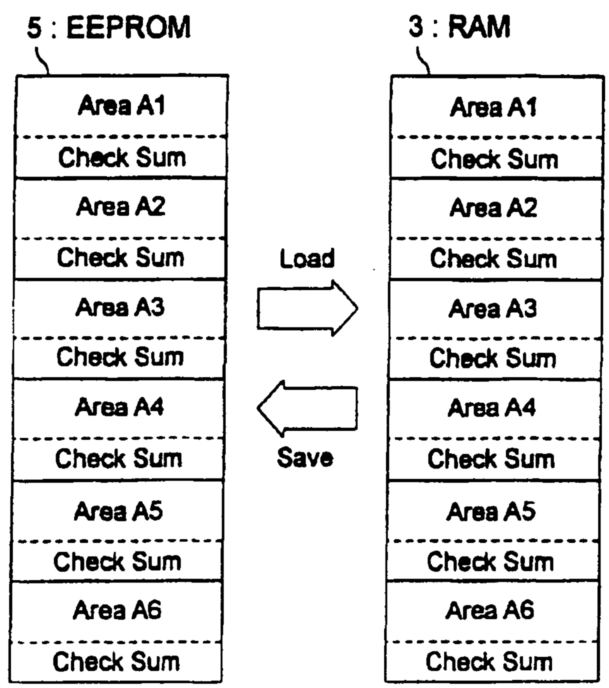

FIG.3

	A1	A2	A3	A4	5A	A6
Printer Power ON	Save					
Lapse Of Saving Period		Save			Save	
Specific Process Executed		Save	Save	Save	Save	
Specific Amount Of Change In Counter Value		Save		Save	Save	
Reset Signal Detected					Save	

FIG.4

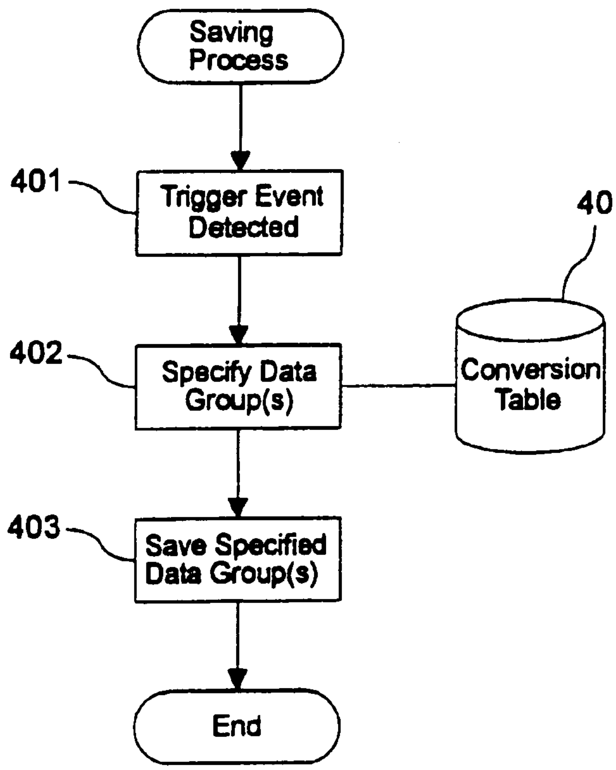
Saving Process
401
Trigger Event Detected
40
402
Specify Data Group(s)
Conversion Table
403
Save Specified Data Group(s)
End

FIG.5

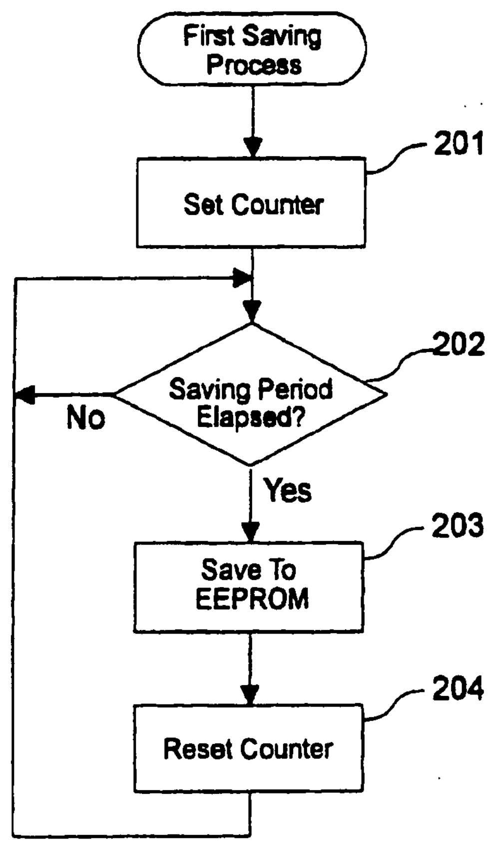
First Saving Process
201
Set Counter
202
Saving Period Elapsed?
No
Yes
203
Save To EEPROM
204
Reset Counter

FIG.6

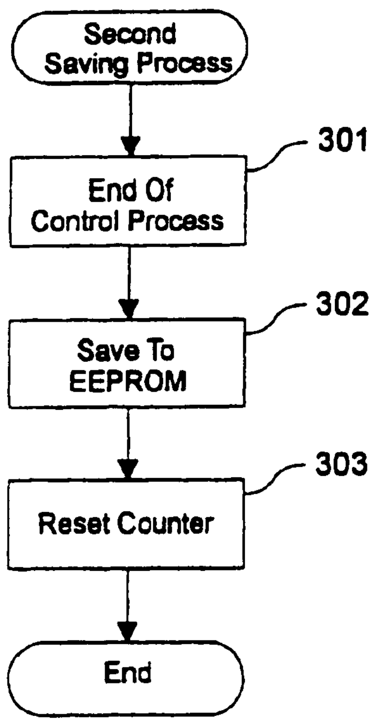
Second
Saving Process
301
End Of
Control Process
302
Save To
EEPROM
303
Reset Counter
End

FIG.7

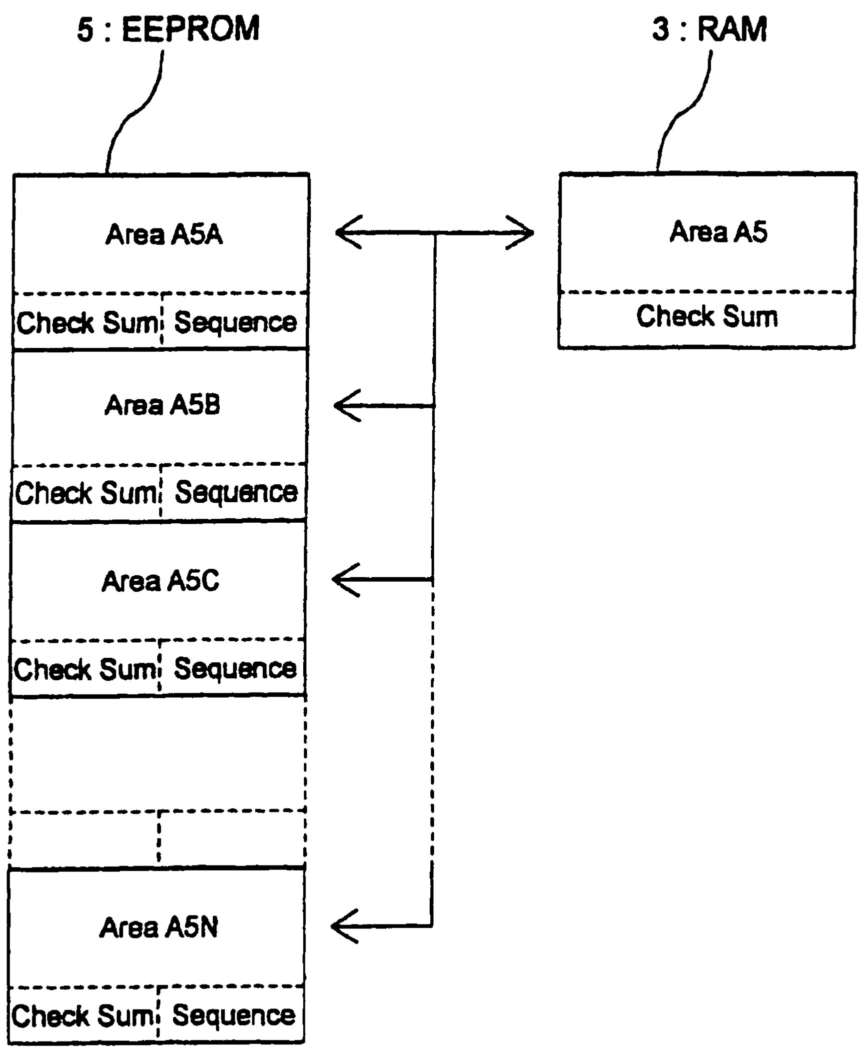
5 : EEPROM
3 : RAM
Area A5A
Check Sum
Sequence
Area A5
Check Sum
Area A5B
Check Sum
Sequence
Area A5C
Check Sum
Sequence
Area A5N
Check Sum
Sequence

FIG.8